Happy Holidays

Toples

LA 99ers COMPUTER GROUP

Newsletter

VOL 5 NO. 12 DEC. 1986

TERRIE'S CORNER

Teresa Masters President

REMINDER, REMINDER, REMINDER
you know what they say is the first thing to go.
December 18, THURSDAY, YES that is the correct day for our December meeting, same time, same place. Didn't think you wanted to share Christmas Eve with us, neither did the Library. So Dec. 18 it is.

HOLIDAY CARE, as is our yearly gesture, PLEASE remember to bring canned or boxed food to join with the Ham or Turkey the Club will buy and contribute to a "Feed the Hungry" organization. We have done this now for a few years and it is a good feeling to participate. Be as generous as you can.

HOLIDAY SHARE, another annual event for this Holiday Season, is to bring along a wrapped computer related gift to exchange during this meeting. There have been some very imaginative gifts exchanged in the past years. He who brings, receives. Wrapped and reasonable.
MORE HOLIDAY SHARE, remember the young lady we were going to "adopt"? We were going to keep her supplied in programs. Well a little reminder Jennifer Johnson, 4886 Doliva Dr. San Diego, Ca. 92117. A little background, Jennifer is 13 and has Cerebral Palsy. The computer is her communication link. She studies in the 3-6 grade level.

Overseas Military Members, we have several, how about looking over our address list and remembering our "away from home members". Got a letter from Paul Sparks today mentioning a "lonely in a personal sort of way" Thanksgiving. Having been a part of this fine group for several years now, I am sure you will find the time to spread some cheer outside of your own circle. Thanks. And speaking of our military members, Paul Sparks sends us an urgent request. He bought an Okidata ML 93 printer and can't figure out the parallel cable for it. The company was singularly unhelpful. If anyone out there can figure this out PLEASE HELP and send it to Paul at SS ELIZABETH LYKES, FPO NEW YORK, 99577-7216.

November Meeting, well for the eve of Thanksgiving, we had a surprising turnout. An excellent program with some very warm vibes from our Members. What a sharing evening that was. How nice and rewarding it is to be a part of this group. Well just a few days later at the Board Meeting, I was overwhelmed at the caring, and sharing in the Club's interest that was going on. It was not just 3 or 4 of us involved, there were at least 18! All contributing in a positive productive manner. L.A. 99er Board is the reason the Club is as productive and alive as it is. The catalyst is the people, everyone likes and respects one another. Ego and jealousy just do not participate. It is a great meld of interested, intelligent, well meaning people. Our Newsletter is the combined effort of several. The Meetings are also, the Library has no equal, Fred Moore is #1. If you have missed meetings lately, make an effort to attend. The Board Meeting is open and we welcome you. You will be contacted by telephone for your needs and views, and offerred the opportunity to actively participate, think it over and join us. We are not the largest Club, we have not advertised for Members, our "FEST" is not in competition with any other. We are simply sharing the best we have to offer, with the best we know. It has really turned into a great two-way Boulevard. My personal Season's Greetings to ALL of you and my gratitude for being one of you.

Coming Soon, An Afternoon with Gram Kracker, hosted by Tom Freeman. There was an amazing show of hands of Members who own Gram Kracker, and at least 8 more who plan to get one! There was a request for "GK made Easy". Well we got the message, and we will do so shortly. We

also plan future "Weekend Afternoon With——" You fill in the blanks and we will find the Expert to Host that day.

99'FEST-WEST'87 May 2, 3. This is a voted on and passed reality now. Invitations to Vendors are being prepared now and will go out right after the Holidays. There have been several commitments already. There will certainly be important new products for all to see. Put this one on your Vacation Calendar, and plan to attend.

L.A. 99er BBS, this too has been voted on and passed. We hope to make this a reality in the very near future. We plan to have a Fund Raiser in order to start at "the top". Dedicated system, hard drive and all. Stay tuned for this. John Williams we hope to draw on your experience in this venture.

Craig Miller and Lou Phillips, the now and the future of our Community. Both taking Giant steps in a forward direction, each a different fork. I know, admire, and personally like each man. I sincerely wish them both well in their path.

Craig Miller, I wonder just how much the Community really realizes what fantastic contributions Craig and Sue have made. Craig's learnings were ours, for a closed architecture machine, Craig shared every opening he discovered. Every program he wrote was a Tutorial for all of us. He gave far more than he received. A few of us recognized this great value and drank hungrily of it. Tom Freeman shared with me his personal thoughts of how very much Craig Miller is responsible for Tom's progress. The many questions freely answered, the information shared, the friendship extended. We, as a Community are where we are because of Craig Miller.

Lou Phillips, For the entire time I have known Lou, he has had ONE direction, FORWARD. One cannot even fairly call it a dream. Lou has known what he is capable of doing, and has steadfastly gone about doing it. How good it is that he can have the concentration of Arnold Palmer, or Jack Niklaus and ignored all the "traps" strewn about. 9640 is a reality. Compare it to what is on the market now. Compare memory, graphics, price. Just compare. Tenaciousness and foward direction. Not a bad combination.

CLUB OFFICERS

President:		Membership Chairman:	
Terrie Masters	(213) 271-6930	Tom Freeman:	(213) 454-1943
Vice President:		Librarian:	
George F. Steffen	(213) 329-3527	Fred Moore	(213) 670-4293
Secretary		Library Assistant:	
Terry Wilson	(213) 563-3869	Chick De Marti	(213) 532-8499
Treasurer		Library Assistant:	
Margaret H. Hutton	(213) 541-5359	Allen S. Whiteman	(213) 379-8031
Editor:		Equipment Chairman:	
Terrie Masters	(213) 271-6930	Joe Fierstein:	(213) 377-9834

FOR SALE

TI PRINTER NEW NEVER USED	$250.00
ANCHOR 300 MODEM " "	$ 40.00
P-CODE CARD	$125.00

ED CADWELL (818) 335-0364

BLACK/SILVER 99/4A	$ 35.00
SPEECH SYNTHESIZER	$ 25.00

TERRIE MASTERS (213) 271-6930

L.A. 99ER'S DECEMBER PROGRAM

The December meeting will begin with the President's report by Terrie Masters. Fred Moore, the club software librarian, will discuss the new additions to the collection. Following Fred's report, Gail Fair will report on the month's Marketplace specials.

Steve Mehr will be demonstrating a new program called PRINTIT. PRINTIT is another program in the groupware concept and Steve will have copies available for those who are interested. The PRINTIT demonstration will be followed by a social break.

The December meeting will begin a new series of demonstrations of communications over the telephone with the TI 99/4A. Many club members have modems and have heard about TE II, FASTTERM, bulletin boards and other things. However most people are unaware of how much fun and how easy it is to get online to another computer. Steve Chalcraft will begin the series starting with the basics. Steve will demonstrate the use of modems and how to connect the phone to the computer. Also the reasons for the communications software will be explained. Questions are encourged.

There will be two special events during the December meeting. First Doug Moore will be conduction a raffle. The first prize will be a years subscription to MICROpendium, the first class magazine about the TI 99/4A. The second event will be the annual drive for canned food or articles for charity. Lets remember other during the holiday season.

The December meeting will be hosted by Fletcher Wicker. Happy Holidays.

DATA BASE MANAGERS by George F. Steffen

I have been looking for a good data base manager for some time. In most cases, I have found that writing an EXTENDED BASIC program to do the job is much more satisfactory than using a manager. My preference among those I have used is Navarone's DBMS because the files are compatible with standard TI files and thus are easily accessible from BASIC or EXTENDED BASIC. I found it impossible, for instance, to produce a report in our desired format for our group library list. Therefore, I set up a DBMS file for entry of the programs and wrote a program to produce the printed report. The DBMS is used for entry and sorting.

My objections to most data base managers are: 1) Because fields are fixed length, variable length data occupies too much space. 2) No way to restrict entered data to a desired range. 3) Lack of calculation ability.

The following two articles on data base managers were extracted from other newsletters. The first, by Bill Gaskill has appeared in The Front Ranger, Northern Nevada Ninety-Niners Newsletter and Chicago Times. None give credit to any other newsletter nor claim that Gaskill is a member of their group. Therefore, the original source is not known. The other article by Jack Tophan is an extract of an article which appeared in Chicago Times. I omitted his comments on ACORN 99 and PRBASE since they were covered by Bill Gaskill.

MERGING PROGRAMS by George F. Steffen

I have recently read articles in various newsletters pointing out that, unless you type a BASIC or EXTENDED BASIC program in without mistake from beginning to end, the code will be scrambled with the last line entered below all previously entered material. While the above is true, these articles go on to conclude that the running of the program is thereby slowed down because the program must search the entire length of the program to find its lines. They therefore recommend saving the program with the MERGE option and then merging it back so lines will be arranged in numerical order.

When a BASIC program is kept in memory, it is in two parts: the line number table and the tokenized statements. The statements are entered, one line at a time, starting at the top of available memory. Below the statements is the line number table, each entry consisting of the line number and the starting address of that line. The line number table is kept in numerical order, with the lowest in the highest memory address.

When a line is changed, the old line is eliminated and anything below that moved up; the new line is inserted below the remaining lines but above the line number table, and the starting address for that line in the line number table is corrected. Since the line number table is always in order, running speed is not dependent on the order of the tokenized statements.

	FEATURE	ACORN99	DBMS	DB1	DB99	DB X	PRBASE	TURBO DM
F	RECORDS/FILE	LIMITED BY DISK	32,000	LIMITED BY DISK	350-1400	LIMITED BY DISK	350/710	LIMITED BY DISK
E	FIELDS/REC.	54	25	10	28	10	32	30
A	MAX. RECORD LENGTH	255	253	245	246	246	246	255
T	MAX. FIELD LENGTH	40	40	28	28	28	246	28
U	MEMORY REQ'D	32K	32K	32K	32K	16K	32K	32K
R	LANGUAGE	XB/ASSM	ASSM	XB/ASSM	XB/ASSM	XBASIC	ASSM	XB/ASSM
E	CUSTOM DESIGN SCREEN LAYOUT	NO	YES	NO	YES	NO	YES	YES
S	SCRN GRAPHICS CAPABILITY	NO	NO	NO	NO	NO	YES	YES
T	ALTERED CHAR SET USED	NO	YES	YES	NO	NO	YES	NO
A	CUSTOM REPORT DEFINITION	YES	YES	YES	YES	YES	YES	YES
B	SAVES REPORT DEFINITION	NO	YES	YES	NO	NO	YES	YES
L								
E								

DATA BASE MANAGERS FOR THE TI-99/4A by Bill Gaskill

Some owners/authors of the applications I have covered in this article will no doubt be angered by the apparent brutality of it. I choose to view it as honesty rather than brutality. Too many reviewers white wash the weaknesses of TI software they critically review. I will not. I think sometimes that we are afraid that the software market will dry up and blow away unless we give favorable reports on the software products that do appear for our computer. I prefer to think of it in another way: if we promote junk software in a favorable light, those that publish product reviews will lose credibility, and those that buy software based upon those reviews will simply be that much more reluctant to get burned a second time.

In the process of searching for the perfect data base manager, I have purchased several programs and spent over $300. All of the programs that I own have positive points and all have negative points. What I have discovered to date is that the "perfect" data base manager does not exist yet (not even in the business world). What I am going to do will perhaps save you a little time and money if you too are looking for that perfect application.

The programs I own are:

ACORN 99 from Oak Tree Systems
DBMS from Navarone Industries
DATA BASE 1 from SPC Software
DATA BASE 99 from Quality 99 Software
DATA BASE 300 from the International Users Group
DATA BASE X from Western Ware
PRBASE V1.2 and V2.0 from William
TURBO DATAMAN from Easy Ware

I have used these programs enough to feel comfortable with each and could probably write several pages about each one. Unfortunately, publication space is limited and such a voluminous article would never see print because of it. Thus I have tried to be brief, but to the point, in my comments on each program. Also, please keep in mind that my comments are subjective, based upon how each product meets MY needs and expectations. Yours may be different.

For ease of reference I have included some of the information in a comparison table that allows analysis at a glance. In the paragraphs that follows I will try to provide a little detail to each issue and cover special features, lack of what I view as standard features and product performance of each program. I apologize in advance for the cryptic style you will read, however, I needed to be brief. The DATA BASE 300 program will not be looked at since it is not available.

ACORN 99:

Among the top three DBM's available to the TI community. The only relational data base available. Also, the only one with a programming language interface for custom applications. EXTREMELY powerful and well designed. Can support three active files at one time. Allows existing data file formats to be edited, copied to another file, resequenced and can reformat a file structure into aother file format. Does not have the ability to show numberof records in a file. Can hold more than 1500 records per file on a SS/SD disk (depending on file size). Sorts alpha characters and strings better than numbers, indexes record location for subfile creation and main file is then concatenated to create the subfile as another database. Posseses ability to search using, "equal to, unequal, greater than, less than, ignore" logical operators. Supports relational operators in search routines through the use of a true/false convention that allows selecton of records where all parameters are met, or any parameters are met. CAN print a single record from a display screeen. EXTREMELY slow in operation. Uses 40 column text mode. Allows duplicate key field data entries. Allows printer control codes to be encrypted in setup file. Provide input checking for "numeric, integer, money, string, flag and date" entries. Overall, a fabulous program, with almost limitless potential. The best documentation of the group, giving many examples along with explanations. SUPERB application.

DBMS (Navarone):

Allows 32,000 records per file, but only 350 per SS/SD diskette. Limits you to half that amount if you wish to sort the file since it creates a second sorted file that demands equal space on your data disk. Most interesting report generator I have ever seen, a cut and paste type affair that is really neat, but poorly documented. Excellent custom screen design module which includes help screens that you design. FAST FAST FAST. Requires unique key field entries only, which I find inconvenient. Documentation is better than originally written, but still confusing at times, and incomplete. Dotes on mundane things and skips over, or entirely omits, important things. Does totalling in reports, but no other computational work. Does not suport single record printing, but can use the report module to scroll data on screen, write it to disk or send it to your printer. Can append new files to the end of an existing record, but cannot reformat the record in any other way. Can create subfiles, but you have to figure out how to do it for yourself because the documentation does not tell you how. It doesn't even mention subfiles. Allows printer control codes to be encrypted in Report Generation file. Does not perform input checking of any type. All data is considered to be a string entry. Best suited for a hard disk environment. Not difficult to use once you have "played" with it, but can be intimidating at first.

DATA BASE 1:

Best suited for mailing lists or other LIST type data files. Cumbersome design setup requiring records to be accessed by their relative position in the file (record number). You must first list the records by a specified

field if you don't know the record number. Time consuming. Provides three pre-set mailing label report formats and one custom format for your own design. Will NOT do reports that have heading information. Includes several nice utilities, such as a formletter generator, disk file data base which creates a DB1 data base file out of the information on your library of disks. Does not provide for input checking, nor length of field entries. Only looks at the length of overall record. Does searches by "equal to" operator only on one data field at a time. Requires that you first create an index file and then search. To search by another field you must create another index file. Searches by a maximum of 5 characters in any field. Sorts are limited to 1000 records, no matter how many exist in the file, but both alpha and numeric sorts are offered. Subfiles can be created to a printer in the main program or to disk by using the Utilities options. Selection is by "equal to" or "between two values", which can be either alpha or numeric type.

DATA BASE 99:

More emphasis put on copy protection than on program performance. Allows custom screen design and claims 28 fields of up to 28 characters each. Would be a neat trick to do since four of the 24 rows on screen are used by program prompts. Fast assembly language interface for report generation. Cannot generate reports with headings and does not permit printer control codes to be inserted in report data. Does not save a format after design, so you will have to re-create it each time you want a report. Data is printed in continuous format without regard to page breaks or anything else. Design of layout is cumbersome, requiring you to conceptualize how many colons and/or semi-colons are needed to push the data across the page. Number of colons/semi-colons is limited to 127 characters allowed in a LINPUT command. A terrible system. Disk catalog accessed from main menu will crash program if you enter an alpha character instead of a number when it prompts for the disk drive number to be catalogged. Color is lost after a crash since it was CALLed from the LOAD program. Does not permit single record screen print (unless you buy the DB99 Utilities), must use EDIT option to search for a record or search sequentially. Cannot go directly to a record by its relative position in the file. Will create subfiles to disk, allowing the search by "less than, equal to or greater than" operators. Search is limited to one field for all practical purposes. Sorts can be performed in ascending order, by any one field. Sort is an actual re-write of the file. All data is considered string information. No number crunching (again, unless you buy the DB99 Utilities), no input checking. Documentation consists of two 8 1/2" X 11" sheets of paper printed on both sides. Program is slow, inflexible, inconvenient in many ways and cumbersome to use. It might have been an advanced application two years ago. Today it is a dinosaur, even with the DB99 Utilities. Much too expensive.

DATA BASE X:

Very modular, meaning that each function (adding, editing, printing, deleting etc.) is a seperate program that must be loaded each time you want to use that function. Does statistical analysis of data. Record counter is inaccurate, code of program is jumbled and entirely unstructured. Does not sort data, even though documentation uses the term "sort". What it means is "select". When DATA BASE X "sorts" by a particular parameter it is really selecting records for dumping to a printer that meet that parameter. Does allow selection between ranges. Cannot create subfiles, does not index existing records. Access of a record is done sequentially, unless you know the record number. No way to tell the record number, you must guess. Supports 1 or 2 disk drives. Excruciatingly slow. Requires that you name the data disk DBXDATA, for no good reason that I can see, otherwise program errors out. Dopes not save report definition, but does allow it to be printed in normal or compressed mode. Definition process if fairly simple, but time consuming. Documentation is the "shabbiest" I have ever seen. It is photocopied and put into booklet form with the pages not even cut straight, so that some information is missing off some pages. Overall, this program is JUNK! As with the IUG's DATA BASE 300/500, it never really belonged on the market in the state that it is in. Unfortunately, I didn't know that and paid over $30 to find out.

PRBASE:

Totally assembly language coded. THE BEST all-around application in my opinion. FAST, flexible, does virtually anything a user would want in the way of data handling, except number crunching. It will not do anything in that area. Treats all data as part of a big string just as DBMS and DATA BASE 99 do. As long as you own the PRB Utilities written by John Johnson you can create subfiles, otherwise you can't. Has on-line help for commands, creates an index by an input field you choose and then accesses any record in about 1 second. Also has a FIND feature to look at data sequentially in any single field and a GLOBAL option that searches for a single data entry anywhere in the record. Saves up to five report formats, V2.0 allows you to format a data disk. Custom screen layout with terrific graphics options for borders/windows etc. is avalable. A tremendous program, well thought out, well designed, artistically executed. FAIRWARE!!! PRB Uitilities are free for the asking as long as you provide th disk and mailer. Report design routines is cumbersome and confusing. Prints single record from screen display in either 40 or 80 colum mode. Program is very sensitibe about I/O device names. My copies (V1.2 and V2.0) both require PIO. to work rather than just PIO or PIO/1, etc. With number crunching abilities this program would be a perfect "flat-file data manager" for most TI users. As it is, te value and performance for a FAIRWARE application, or a commercial application too for that matter, is unsurpassed. If you don't have PR BASE then you are missing out on one of the premier productivity tools available to the TI Community.

TURBO DATAMAN:

This is the second most powerful and useful data manager, taking a backseat only to PR BASE. It runs slightly ahead of ACORN because it performs number crunching and is faster in operation. Like ACORN, TURBO DATAMAN allows you to create a dictionary of data items (fields) and then lets you choose from that library of fields to put a record together. Up to 30 fields are allowed per record. Twenty pre-defined records (file formats) can exist on one disk. Allows custom screen layout design, complete with graphics for borders/windows etc. Does input checking, allows secondary screen access, like ACORN's Detail Records. Alows formulas to be created and saved that perform the four basic math functions. Report definitions can be saved. Allows wildcard type operators in searches, will print single record from screen display. Provides "less than, greater than, equal to, not equal to, greater than or equal to, less than or equal to" operators in screen display and report generation modules. Permits sub-totals in reports that can be formatted like TI Extended Basic does with the IMAGE statement. Subfiles can be created through the report generator by sending the output selected to a disk file rather than a printer. The results must be converted back to INTERNAL, FIXED from DISPLAY FIXED before you can use it in the program, however. TURBO DATAMAN does not provide you with that utility. The documentation instructs you to "write a program" to do it. Names used for different modules in the program are confusing. Eg: ETCH, SKETCH, SKETCHR, FETCH. Should change names to more accurately reflect function of module. Documentation acceptable,but lacks adequate coverage in some areas. Utilities are provided to perform some mundane operations, such as counting the amount of records in a database. Reformatting or restructuring of an exixtng file is not permitted, unless the input field is appended to the end of a record format. This program needs some "fine tuning" in some areas, but is still an exciting productivity tool wth immense possibilities. Its speed of operation is not fast, but acceptable. It is faster than ACORN. One can set up the SKETCH program to auto-load if desired, but the whole application should be centered around a menu, in my opinion. As it is now, you must RUN each module from the READY> prompt when you need to use it, because every module exits with an END statement. If you don't own this program, you should. Whether you want to manage a mailing list or do accounting, TURBO DATAMAN is for you.

SOFTWARE REVIEWS: Jack Tophan

Back in May 1985, I reviewed several DATA BASE MANAGEMENT programs and concluded that ACORN99 was the one to beat. Indeed, ACORN99 has remained one of the few really complete data base managers available for the TI99. Since it has remained my benchmark program, I will review its features once again.

...

Now along comes two new FreeWare programs from two different parts of the TI99 world. I'll start with CREATIVE FILING SYSTEM from Mark Beck in ARKANSAS. CFS has 24 pages of documentation as well as a set of example files that the DOCS walk you thru step by step. A nice touch! CFS requires a full system and can be loaded using any of the load modules. Two drives or a lot of disk swapping is required. Data goes on one and the CFS files on another. This permits auto load of data files which I find great. Up to 1400 8 line records can be created on one DS/DD disk. Each line can have its own printing default characters. Sub files can be created on user defined criteria. Mailing labels can be printed up to 4 across. Any number of files can be merged together and files can be converted to DV/80 to be used with TIWRITER. Any file can be converted to a TIWRITER VALUE file as well. Mathematical operations can be performed in the REPORT mode. I trust I have whetted your appetite by now.

RECORDS can be up to 16 fields of 14 chaacters each and are defined in the CREATE mode. With up to 16 field headings of 14 characters each, the record screen is 16 lines. Once the fields are created, you can enter data record by record. Once entered, you can display the files in several ways. Start to finish, from start at to end at, selected files, or letter to letter in any field. Sub files can be created n this manner as well. The APPEND mode allows adding more records to a file.

In the SEARCH mode single or multiple conditions may be used. Since every field is searched, it takes a short while. If any one field only is searched, the search is fast! The CHANGE mode allows records to be edited for changes on a singel record or on all records. Up to 1300 records can be sorted on any one or two fields. Even a variable sort is offered if the data is in different fields in different records.

A unique option is GRAPH. If numeric data in one of the fields is preceded by a $ sign and a date is in another field, CFS will display a screen graph in-color and/or print out the data. Two graph options are offered.

CFS was designed to use FOUNDATION's 128K card if you have one to increase the speed of SORT and GRAPH. NOTE the files "DISK" and/or "PRINTER" delete them before starting. They will be created uniquely for your system after loading the first time. CFS will print out reports in various ways including mathematical operations on the data. For example, add col 3 and 4 for each line and print total in col 5. Or totalling can be done vertically. And fnally, CFS will let you CATALOG 20 disks of up to 50 files each and then use all of CFS utilities to SORT, SEARCH and REPORT. If you like CFS send $10 to MARK at 166 DELAWARE CIRCLE, JACKSONVILLE, AR 72076.

Another hot data base these days is PRBASE, A PERSONAL RECORD MANAGEMENT SYSTEM BY William Warren, ...

MARKETPLACE

(the marketplace is a fund raiser for the club, that is, the "profit" goes to maintain the quality of this Newsletter. In general the price listed splits the difference between cost and retail. Please help your Club.)

MILLERS GRAPHICS	
DISkASSEMBLER	18.50
ORPHAN CHRONICLES (priceless)	9.95
ADVANCED DIAGNOSTICS	18.50
EXPLORER	22.50
NIGHT MISSION	18.50
GRAM KRACKER (80K EXPANDED)	185.00
GK UTILITY I	10.00
SMART PROGRAMMING FOR SPRITES	6.25
NEW RELEASES	
PRE-SCAN IT! (J.PETER HODDIE)	10.00
GRAM PACKER "	10.00
FONT WRITER "	19.00
PRINTER'S APPRENTICE (M.McCANN)	19.00
MYARC	
RS232	82.00
D/D DISK CONTROLLER	155.00
128K RAM DISK/SPOOLER	175.00
512K RAM DISK/SPOOLER	280.00
EXTENDED BASIC II LEVEL IV	80.00
128K RAM DISK W/XBASIC II	235.00
512K RAM DISK W/XBASIC II	340.00
INSCEBOT	
TI-ARTIST	15.00
DISPLAY MASTER	12.00
ARTIST EXTRAS	6.00
MEGATRONICS	
EXTENDED BASIC II PLUS	72.50
INTERN (BOOK ON GPL)	16.50
128K GRAM CARD	227.50
HARDWARE & SUPPLIES	
TEAC 55BV DSDD DRIVES	90.00
DISKETTES DSDD	.50
64K EPSON INT. PRINT BUFFER	45.00
COLOR RIBBONS (EPSON)	4.00
BACK ISSUES	
SUPER 99 MONTHLY	1.25
MICROPENDIUM	1.25
SMART PROGRAMMER JUNE 1986	1.50
BEST OF NEWSLETTERS W/DISK	5.00
FORTH NOTES VOL 1-5 (2.50 EA)	10.00
BEGINNER'S FORTH NOTEBOOK	2.50
ASSEMBLY NOTES VOL 1	2.50
TECHNICAL AND BUSINESS BOOKS	5.00
SAMS BOOKS (VARIOUS)	5.00
SAMS BOOKS WITH CASSETTES	7.50

*** MYARC TO CORCOMP DSDD DISK CONVERSION ***

by

Sid Smart and Jim Lohmeyer

THANKS TO "MICRO"

We returned from the Chicago TI Faire with some swapped disks that couldn't be read with a Corcomp disk controller without errors of one kind or another. Some would catalog (showing 1280 total sectors) and some wouldn't. Reading sectors with Millers Graphics' Advanced Diagnostics revealed a pattern of 16 good sectors followed by 2 "bad" sectors. Of course, we had Myarc DSDD disks with 16 sectors per track, and a Corcomp controller expecting 18 sectors per track. The Myarc controller that wrote the 17th and 18th sectors on the disk thought they should be the 1st two on the second track. The Corcomp controller reading the disk thought that the 17th and 18th sectors should be at the end of the first track! So all we had to do was move them: read 16 sectors, write 16, skip 2, read 16, write 16, skip 2, etc. Advanced Diagnostics will handle it, but that's too much typing to enter in immediate mode for even one such disk! Fortunately, AD can be driven with a command file, and the commands required are repetitive enough that they can be generated from a relatively simple BASIC program. The necessary commands won't fit in one 2K command file, so the program below creates two, and the first invokes the second. Both the comand file generator program below and the command files themselves provide instructions for their use.

```
100 REM ********************
110 REM ** MYARC DS/DD    **
120 REM **      TO        **
130 REM ** CORCOMP DS/DD **
140 REM ** CF GENERATOR  **
150 REM **    11/3/86     **
160 REM **   SID SMART    **
170 REM **      AND       **
180 REM **  JIM LOHMEYER  **
190 REM **LEROY, ILLINOIS**
200 REM ********************
210 CALL CLEAR
220 PRINT "THIS PROGRAM GENE
RATES TWO ";"COMMAND FILES
FOR USE WITH ";"MILLERS GRA
PHICS ADVANCED  ";"DIAGNOST
ICS. WHEN INVOKED "
230 PRINT "(WITH A CORCOMP C
ONTROLLER) ";"THEY CONVERT A
 16 SECTOR PER";"TRACK MYARC
 DSDD DISK TO A";"CORCOMP 18
 TRACK PER SECTOR "
240 PRINT "DSDD DISK.
          ";"THE FIRST COM
MAND FILE IS ";"'MYARC/CC' A
ND IS TO BE  ";"INVOKED BY
THE USER.  THE "
250 PRINT "THE SECOND IS 'MY
ARC/CC2'.  ";"IT IS INVOKED
BY THE FIRST";"COMMAND FILE.
"
260 PRINT : : :"PRESS ANY KE
Y TO CONTINUE"
270 CALL KEY(0,K,S):: IF S=0
 THEN GOTO 270
280 CALL CLEAR
290 FILE$(1)="DSK1.MYARC/CC"
300 FILE$(2)="DSK1.MYARC/CC2
"
310 T1$="SD 1 CR "
320 T2$=" 16 SD 2 CW "
330 T3$=" 16 [13]"
340 T4$(1)=" [7] PA CF DSK1.
MYARC/CC2 [13] []"
350 T4$(2)=" []"
360 MH1$="[255][7]CC 2 8 7 1
3     [13][7]"
370 MH2$="[253][253][253][25
3][253][253][253][253][253][
253][7]PA[13][7]"
380 M$(1)="[7]BEEP[13][7]Pla
ce Myarc disk in drive[32]on
e and press a key"
390 M$(2)="[7]BEEP[13][7]Pla
ce Corcomp disk in dr.[32]tw
o and press a key"
400 M$(3)="[7]BEEP[13][7]Pla
ce #CF# disk in drive[32]on
e and press a key"
410 M$(4)="[7]BEEP[13]CONVER
SION COMPLETE "
420 DISPLAY AT(12,1):"PLACE
DISK FOR COMMAND FILES"
430 DISPLAY AT(14,9):"IN DRI
VE ONE"
440 DISPLAY AT(16,6):"AND PR
ESS ANY KEY"
450 CALL KEY(0,K,S):: IF S=0
 THEN GOTO 450
460 FOR L=1 TO 2
470 DISPLAY AT(19+L*2,3):"CR
EATING ";FILE$(L)
480 OPEN #1:FILE$(L),DISPLAY
,VARIABLE 80
490 PRINT #1:MH1$
500 PRINT #1:M$(1)
510 PRINT #1:MH2$
520 PRINT #1:M$(2)
530 PRINT #1:MH2$
540 FOR I=I+1 TO I+48
550 R=(I-1)*18 :: W=(I-1)*16
560 L$=T1$&STR$(R)&T2$&STR$(
W)&T3$
570 PRINT #1:L$
580 NEXT I
590 IF L=2 THEN 700
600 PRINT #1:M$(3)
610 PRINT #1:MH2$
620 PRINT #1:T4$(L)
630 CLOSE #1
640 I=I-1
650 NEXT L
660 CALL CLEAR
670 PRINT "LOAD ADVANCED DIA
GNOSTICS,  ";"PUT COMMAND FI
LES IN DSK1 ";"AND ENTER THE
 COMMAND:      "
680 PRINT : :"CF DSK1.MYARC/
CC": : : : : :
690 STOP
700 PRINT #1:M$(4)
710 PRINT #1:MH2$
720 GOTO 620
730 END
```

Did you know that...?

by Chick De Marti

So simple, Why didn't I think of it?

```
1 REM **** 10 RND NUMBERS ****
5 REM
10 FOR X=1 TO 10
20 LET N(X)=0
30 NEXT X
40 FOR I=1 TO 10
50 R=INT(RND*10)+1
60 IF N(R)>0 THEN 50
70 N(R)=R
80 PRINT R,
90 NEXT I
100 END
```

CTRL S = DATA

I "discovered" the time saving routine of using FCTN 8 (REDO) when entering long lists of DATA. Exam>:
100 DATA aaa,bbb,ccc,ddd <ENTER>
FCTN 8 brings upa a duplicate line of 100. I use the arrow(->)key to change 100 to 110,skip over the word DATA and now just enter my new data:
110 DATA eee,fff,ggg,hhh

Now! I "discovered" an easier system!! Type NUM <ENTER>. At line 100 enter CTRL S, space bar, and enter the data. That's all there is to it! The CTRL S will show a blank, but when listed..it displays "DATA". Easy, huh?
An APPLE student of mine likes to show at TI use CTRL : for the same thing!

Oops...

Last month's TRIVIA IN NUMBERS answer was transposed. (Thanks George Steffen for finding it.) The correct answer is for finding it.) Here it is corrected.

What 3 different digits are represented by A,B,C in this addition problem?

```
  A C B
  A B C
  -----
  B C A    (is the correct answer.)
```

FORTH through Mini-mem?

Yes, you can load FORTH using Mini-mem
Use Option 3 "Minimem" then...
Option 1 "Load & Run"

And thanx to...

Boston Computer Society TI-99 User Group

By Aaron W. West

Has your computer ever locked up while saving an Editor/Assembler file? (Ours did when I turned off a light with our Percom Data drive running.) Try to avoid spikes, and maybe use a spike suppressor, but if your computer does lock up type in CALL LOAD(-31860,96,41) in TI BASIC to return to E/A without reinitializing the memory expansion. You can also use this to rerun a program that you loaded in E/A. [Note: this will only work with an E/A Cartridge located at GROM address >6000. This will not work with a moved E/A using a GRAM Kracker. -jph]

Random Bits from Bill Wallbank

Here is a non-fatal solution to determine the language enviornment...

```
100 REM  Steve Chapman & Bill Wallbank
of Stone & Webster Engineering Corp.
TIUG
110 RANDOMIZE (0)
120 V=INT(RND*100)
130 IF V<>21 THEN 180
140 V$="Extended BASIC" :: GOTO 190
180 V$="TI BASIC"
190 PRINT V$;V
```

Since this routine might be RUN from another XB program, the random sequence MUST be reset by using a contant seed as in 110. The random number generators return different sequences, using a seed of 0: BASIC returns 82, XBASIC yields 21. By not attempting to execute a line containing invalid tokens, no interpreter problems.

STOP SCROLLING??

Want to stop scrolling by holding down the space bar (or any key)? This neat routine is from READ/PRINT D/V 80 FILE by W.A.Wilson (Oct. issue San Diego TI SIG Newsletter).

```
260 LINPUT #1:X$
270 IF EOF(1)THEN 360
280 PRINT X$
 ^
320 CALL HOLD
330 GOTO 260
 ^
360 CLOSE #1
 ^
```

```
450 SUB HOLD
460 CALL KEY(0,K,S):: IF S=0
THEN 480
470 CALL KEY(0,K,S):: IF S=-
1 THEN 470
480 SUBEND
```

ANOTHER PROGRAMING HINT
(Who wrote this ???

When working on a program, you save it to disk often just in case your system locks up, etc. To save time, use a... working name of <A> for these frequent saves. This saves up to 9 keystrokes. Also...when cataloging...your working program will be at the top of the list

<*><*><*><*><*><*>

This suggestion by Art Byers appeared in the CALL SOUNDS Newsletter. (I put it in a program form)

```
1 ! SAVE DSK1.nnnn
20000 SUB ANYKEY
20002 DISPLAY AT(24,2):"<Pre
ss any key to continue>
20004 CALL KEY(0,K,S):: IF S
=0 THEN 20004
20006 SUBEND
20008 SUB HONK
20010 CALL SOUND(300,-3,3)
     20012 SUBEND
20014 SUB WAIT(D)
20016 FOR I=1 TO D :: NEXT I
     20018 SUBEND
```

Then save as:
SAVE DSK1.PROG/START,MERGE

Thanx News Net 99er Newsletter
JUST WHEN YOU THOUGHT YOU HAD EVERY-THING FOR YOUR COMPUTER! A,T&T has a compact flat (tone or pulse) phone that is color coordinated with the black and silver console. PANASONIC has one for the beige console.

<*><*><*><*><*>

A TICO TOPICS page entitled "RANDY'S RUMOR RAG" had this article:

<<CHEAP TRICKS>>

CorComp says that you can't load thier manager from within a program. Lee Bendick says you can. Here's how

```
100 CALL INIT
110 DELETE "LD-CMDS"
120 CALL LINK("MGR")
```

<*><*><*><*><*>

If YOU have any "I Didn't Know That!" contributions or stoeies, send them to:
Chick De Marti
P.O.Box 3547
Gardena CA 47247-7247

Send 'em in any shape or form. I'll use it!

Out of coffee. See you next month. Chick

KIDS ##########

Speak to Me [Part 4]

by

Bonnie L. Snyder

This article is a step by step tutorial on how to include Text to Speech in an already existing Extended Basic program. The program I chose is called "Trivia Quiz" and although there was no indication in it as to its authorship, I believe it was written by C. Regena. It runs equally well in Basic and Extended Basic [it appears that it was originally written in Basic hence the mix you will see of PRINT statements which were already in the program, and DISPLAY AT statements which I added where necessary]. In order to arrange the program in Extended Basic form, I ran it through the Compactor program which compresses the listings into multiple statement lines. Next, I listed the program to the printer so that I could see what I was working with. Here is what the base program looks like:

```
100 REM TRIVIA QUIZ
110 DIM S$(20),A$(20):: N=20 :: CALL CLEAR :: PRINT TAB(8);"TRIVIA
QUIZ"
120 PRINT : :"A QUESTION WILL BE SHOWN." :: PRINT : :"TYPE THE ANSWER
(WITHOUT" :: PRINT :"COMMAS) THEN PRESS <ENTER>." :: PRINT : :"THE
CORRECT ANSWER IS SHOWN ." :: PRINT : :"PRESS THE SPACE BAR TO"
130 PRINT :"CONTINUE." :: FOR C=1 TO N :: READ S$(C),A$(C)
140 NEXT C :: PRINT : : :"PRESS ANY KEY TO START."
150 CALL KEY(0,K,S):: IF S<1 THEN 150
160 FOR C=1 TO N :: CALL CLEAR :: RANDOMIZE
170 R=INT(N*RND)+1 :: IF S$(R)="" THEN 170
180 PRINT S$(R): : :: CALL SOUND(100,1497,2):: INPUT B$ :: PRINT
:A$(R)
190 CALL KEY(0,K,S):: IF K<>32 THEN 190
200 S$(R)=""
210 NEXT C :: CALL CLEAR
220 REM PUT QUESTIONS HERE
230 DATA WHO LOST THEIR MITTENS?,THE THREE LITTLE KITTENS
240 DATA WHO WERE THE THREE MEN IN A TUB?,"THE BUTCHER, THE BAKER, THE
CANDLESTICK MAKER"
250 DATA WHEN WILL THE CRADLE ROCK?,WHEN THE WIND BLOWS
260 DATA WHAT DID JACK AND JILL FETCH?,A PAIL OF WATER
270 DATA WHO CUT OFF THE MICE'S TAILS?,THE FARMER'S WIFE
280 DATA HOW MANY BAGS OF WOOL DID THE BLACK SHEEP HAVE?,THREE
290 DATA WHO VISITED THE THREE BEARS'HOME?,GOLDILOCKS
300 DATA WHO CHOKED ON AN APPLE?,SNOW WHITE
310 DATA WHO SANG FOR HIS SUPPER?,LITTLE TOMMY TUCKER
320 DATA WHAT DID THE THREE PIGS USE TO BUILD THEIR HOUSES?,"STRAW,
STICKS, BRICKS"
330 DATA WHO USED HER LONG HAIR TO SEE THE PRINCE?,RAPUNZEL
340 DATA WHAT DID JACK FIND IN HIS CHRISTMAS PIE?,A PLUM
350 DATA WHAT DID LITTLE MISS MUFFET EAT?,CURDS AND WHEY
360 DATA WHO WAS LITTLE RED RIDING HOOD GOING TO VISIT?,HER
GRANDMOTHER
370 DATA WHAT COULDN'T JACK SPRAT EAT?,FAT
380 DATA WHO STOLE A PIG AND AWAY DIDRUN?,"TOM, THE PIPER'S SON"
390 DATA WHO JUMPED OVER A CANDLESTICK?,JACK
400 DATA WHO SLEPT WITH HIS STOCKINGSON?,MY SON JOHN
410 DATA WHOSE FOOT FIT THE GLASS SLIPPER?,CINDERELLA
420 DATA WHAT KIND OF MEAT DID ONE OFTHE LITTLE PIGGIES HAVE?,ROAST
 BEEF
430 END
```

[Continued next page]

(To make this a speaking program, Bonnie added the program lines listed below. They include a title, and an option to bypass the instructions.
NOTE: lines 1,2,3 and 3990,4000,4010 and 4020 are the actual TEXT to SPEECH routine in XBasic. Ed.)

the very beginning. Once the user is familiar with the instructions. he does not need them any more and can go right into the quiz. Note the SPS variables are always followed by GOSUB 4000 to take the string information through the speech translation subroutine. Lines 132-140 translate the directions into speech and get the program ready to display the questions. In line 180 the trivia question is translated into speech - SPS=SS(R); the user's answer is spoken - SPS=BS (so he can hear what he wrote); and the computer's answer is spoken - SPS=AS(R). Finally, lines 3999 through 4020 are the remainder of the merged LD program to which the program goes every time an SPS needs to be spoken. Here are the changes which were added to the final program:

```
1 CALL INIT
2 CALL LOAD("DSK1.SPEAK","DSK1.XLAT","DSK1.SETUP")
3 CALL LINK("SETUP","DSK1.DATABASE")
110 DIM SS(20),AS(20):: N=20 :: CALL CLEAR :: DISPLAY AT(3,8):"TRIVIA
QUIZ" :: DISPLAY AT(22,5):"INSTRUCTIONS? (Y/N)"
115 SPS="TRIVIA QUIZ" :: GOSUB 4000 :: SPS="INSTRUCTIONS? Y.  OR.  N."
:: GOSUB 4000
116 CALL KEY(0,K,S):: IF S=0 THEN 116 :: IF K=89 THEN 119 :: IF K=78
THEN 135
119 CALL CLEAR
120 PRINT : :"A QUESTION WILL BE SHOWN." :: PRINT : :"TYPE THE ANSWER
(WITHOUT" :: PRINT :"COMMAS) THEN PRESS <ENTER>." :: PRINT : :"THE
CORRECT ANSWER IS SHOWN"
130 PRINT : :"PRESS THE SPACE BAR TO": :"CONTINUE THE QUIZ."
132 SPS="A QUESTION WILL BE SHOWN.  TYPE THE ANSWER WITHOUT COMMAS
THEN PRESS _ENTER." :: GOSUB 4000
133 SPS="THE CORRECT ANSWER IS SHOWN." :: GOSUB 4000 :: SPS="PRESS THE
SPACE BAR TO CONTINUE THE QUIZ." :: GOSUB 4000
135 FOR C=1 TO N :: READ SS(C),AS(C)
140 NEXT C :: CALL CLEAR :: PRINT : : :"PRESS ANY KEY TO START." ::
SPS="PRESS ANY KEY TO START :: GOSUB 4000
180 PRINT SS(R);; SPS=SS(R):: GOSUB 4000 :: CALL SOUND(100,1497.2):
INPUT BS :: SPS=BS :: GOSUB 4000 :: PRINT :AS$R):: SPS=AS$R):: GOSUB
4000
3990 STOP
4000 CALL LINK("XLAT",SPS,BS)
4010 CALL LINK("SPEAK",BS,43,128)
4020 RETURN
```

It's not really that difficult to do. Of course, each program has its own ways of displaying information on the screen and listing it out and looking for those places will help you determine where to add the speech.

If you have any questions, don't hesitate to call and ask. Talk to you next time.

* BONNIE *

From the Front Ranger November Newsletter...Thanx

NOTE: A suggestion from Bonnie, once you have the program loaded, use RUN 100 thereafter (to save reloading the TEXT to SPEECH routine again each time. Ed.

PILOT

Here are three short Demo progerams created by Bill Harms and myself. They are meant to demonstrate what the various commands do, and some suggestions as to how to apply them. Chick

```
R: ON DISK AS: BITMAP
*START                                  ! Actually a User Subprogram
IG:                                     ! Initialize Graphics
T: HERE'S A CIRCLE AT 100,100 WITH A RADIUS OF 10
C: #N<-10                               ! gives #N the value of 10
LP: 8                                   ! starts a LooP of 8
DC: 100,100,#N                          ! DrawsCircle size 10
C: #N<-#N-1                             ! " #N=#N-1 "
EL:                                     ! End of Loop
T: NOW FOR A TRIANGLE
T: LINE FROM 10,30 TO 100,30 THEN TO 100,200 AND BACK
DL: 10,30,100,30                        ! DL: = DrawLine
DL: 100,30,100,200
DL: 100,200,10,30
T: NOW FOR A RECTANGLE. 5,130 TO 90,250
DR: 5,130,90,250                        ! DR: = DrawRectangle
DR: 7,137,95,252
LP: 100                                 ! (these 2 lines create a
EL:                                        delay )
IT:                                     ! Initilize Text
*REENTER                                ! A 2nd User subprogram
T: PRESS >A< FOR AGAIN
T: PRESS >S< TO STOP
AS:                                     ! asks for a Single Answer
MJ: A                                   ! if "A" is Matched ...
J: *START                               ! Jump to subprogram *START
MJ: S                                   ! if "S" is Matched ...
T: Well its been fun. Bye               ! Adios
E:                                      ! END
J: *REENTER                             ! take care of any other key
```

NOTE: To keep final message visable long enough to read add A: just before the last "E:". (A: = input)

* * * * * * * * * * * * * * * * * * * *

```
R: ON DISK AS: P46AVGS
*START
CH:
T: This program will find the average of entered numbers.
T: Enter "9999" to stop entering numbers.
C: #S<-0
C: #N<-0
*MORE
T: NEXT NUMBER PLEASE
T: (Don't just hit enter)
```

[Continued next page]

```
A: #A
M:
JY: #A
M: 9999
JY: *AVERAGE
C: #S<-#S+#A
C: #N<-#N+1
J: *MORE
*AVERAGE
C: #A<-#S/#N
T: The average of #N
T: entries is: #A
T: Press M for More!
T: Press any key but Enter to Stop
AS:
M: M
JY: *START
```

☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆☆

```
R: *** SCREEN LOCATIONS in the
R: *   graphic mode, using DC:
R: *
IG:
GC: 7,8
T: at  30,40 -  30,120 -  30,200
DC: 30,40,10
DC: 30,120,10
DC: 30,200,10
A:
GC: 13,8
T:    70,40 -  70,120 -  70,200
DC: 70,40,10
DC: 70,120,10
DC: 70,200,10
A:
GC: 14,8
T:   110,40 - 110,120 - 110,200
DC: 110,40,10
DC: 110,120,10
DC: 110,200,10
T:
T: Press <ENTER> to quit
A:
E:
```

```
! Same as REM


! Initialize Graphocs
! color Graphics RED on CYAN
! TYPE 1st line...
! DrawCircle..Top left of screen
! Another...Top center of screen
! And one...Top  right of screen
! Used to wait for a key (ENTER)
! Graphics now GREEN on CYAN
! TYPE 2nd line...
! DrawCircles at center of screen


! Graphics to MAGENTA on CYAN
! Wait again
! TYPE 3rd line...
! DrawCircles near the bottom


! TYPE a blank line
! TYPE "quit" instructions
! Wait for <ENTER> to be hit...
! and END program.
```

NOTE: At the "PROGRAM IS DONE" message, to run another program enter PILOT. If this doesn't work , enter -PILOT. (Because you are in FORTH you can also type MON to leave.

Chick

What's new?

Printer Codes Control

Here is a help-mate for experienced programmers and beginners. The following program lines: accept a string of printer codes to activate some special print style, ie. line feed of 36/72nds.(1/2 inch), then analyze that string of codes, ie. 27;A;36 to send the printer the codes in a format it will accept. You don't have to enter CHR$(27);"A";CHR$(36).

I used this routine as the core of a slightly larger program (FAST-PRINT) to: 1> allow a person to test printer code strings in order to find the ones that work, 2> use the output file of the prog. in TI-Writer as a set of Transliterates for general use, 3> merge the prog. parts into existing progs. to allow the user to enter his particular printer control codes for special printing and have the info. stored in a default file. The full set of small mergable or run stand-alone. progs. are in the club library now. It's the 5th in the "FAST" series of prog., following FAST-SCREEN (a Simple block graphic Screen Maker), FAST-CALC (a Window Calculator), FAST-MAIL (a TI-Writer based Name & Address/Labels data base), and FAST-TRAN (a Checkbook Recapper and 12 month spreadsheet-like Schedule for actuals and budgets). This last one really suffered from the lack of a way for the user to enter special printer codes as it was specialized to a Gemini 10X for Compressed or Elite.
By the way, I called them FAST because they speeded-up some process and were quicker than most other ways I knew of - to do the job.

The prog. rejects any number that is less than 0 or more than 255 because that's all the ASCII values there are(I think). It also rejects any F$() having a LENgth of more than 3 digits. In both cases the user just reenters A$.

```
20000 OPTION BASE 1 :: DIM(P(14).F$(15)
20002 CALL CLEAR
20010 DISPLAY AT(14,1):A$
20028 ACCEPT AT(14,1)SIZE(-28):A$
20030 IF A$="" THEN 20028
20032 DISPLAY AT(16,1):"Char. for Transliterate: ":C$ :: ACCEPT AT(16,26)SIZE(-1
      )BEEP:C$ :: IF C$="" THEN 20032
20033 CALL SOUND(100,500,0)
20034 DISPLAY AT(18,1):"P=Proceed          R=Redo" :: CALL KEY(0,K,S):: IF S<1
      THEN 20034
20035 IF K=80 OR K=112 THEN DISPLAY AT(18,1):"" :: GOTO 20039 ! proceed
20036 IF K=82 OR K=114 THEN DISPLAY AT(15,1):"" :: GOTO 20028 ! redo
20037 GOTO 20034 ! BAD INPUT
20038 ! to zero arrays
20039 FOR A=1 TO 14 :: P(A)=0 :: NEXT A
20040 FOR A=1 TO 15 :: F$(A)="" :: NEXT A
20041 L1=LEN(A$)
20042 ! *******get position of each semicolon
20043 FOR A=1 TO 14
20044 IF P(1)=1 THEN 20098
20045 IF A>1 THEN 20048
20046 IF P(A)=1 THEN 20098
20047 P(A)=POS(A$,";",1):: GOTO 20049
20048 P(A)=POS(A$,";",P(A-1)+1)
20049 IF P(A)=0 THEN 20054
```

Program, Docs and Demo available through our library. Under FAST-PRINT

(Continued next page)

KRACKERSNACKS

by Tom Freeman

1. GRAM PACKER

Let me state this right at the outset - this new program by J. Peter Hoddie is what I've been waiting for ever since I got my GRAM KRACKER one year ago. The fine programs that came with the Gram Kracker, and the utility disk that was released recently allowed us to move TI-Writer and/or Editor/Assembler to the Grams that we wished, or to combine them, but this still didn't take full advantage of the copious amounts of GRAM available. Now with Peter's help we can finally do it!

First a brief explanation of what GRAM PACKER can do. Once the main program "GP" is loaded (by Load Module of the GK, option #3 of E/A, or #3 of TIW) there are three main "branches" depending on what it is you want to do. First of all you can load a complete E/A #5 type program into the GRAM of your choice. Along with this clone "GP" also inserts a short GPL loader, and information for the main TI menu screen. If you then choose this item from the main menu, the loader transfers the program back out to CPU where it belongs, and then gives control to the transferred program. This takes up quite a bit of the GRAM (one full GRAM for each 33 sector segment, and there is a chance that this won't be quite enough) but boy does it start up fast! DM1000 takes up two GRAMS, FAST-TERM one, for instance.

Your second choice is to insert a GPL loader that merely takes the program off the device you have specified. What actually happens is that a loader equivalent to the E/A #5 loader, is transferred to CPU, and this then takes over the job of accessing the program on disk. There is no advantage of speed here, unless you have a RAMdisk, as I do, but you save the time of having to type in the program name, and you don't even need to have E/A in place. And it IS nice to see these on the menu.

For both of these 2 choices, you can also assign subprogram names, and DSR names so that from Basic or Extended Basic (assuming they don't interfere with your programs in terms of GRAM locations) you can CALL xxx or OLD xxx to access the programs. This is a nice feature.

The final "branch" is an equivalent loader for GRAMS themselves, essentially replacing that part of the Gram Kracker (so you don't have to go back to Loader On). As an example, when I use SBUG6, my XBasic is destroyed, so I have an item on the menu to reload the appropriate RAMbank for XBasic, and reconstitute it. There are several additional files on the disk that substitute for the E/A #5 loader, or the GK module loader, in a more general way, i.e. when the program runs you must type in the name of the file to be loaded, but I found these less useful, as my E/A is always in place, and of course the GK loader is there if I need it. There are also some files for use with Mini-Memory, and a sample source file for GPL utilities equivalent to the ASL utilities.

Now to be a little more specific as to how the program works. First you load the file GP as stated above. That part is easy! As soon as the program runs you are asked which GRAMS to begin and end with, and then you are off and running. Next you are asked to input the filename, e.g. DSK1.UTIL1, if that is the name of the program you want to load, and the device where it is PRESENTLY located. Then you are requested to assign a name to appear on the menu - this is mandatory. Next you are given the choices to choose subprogram or DSR names, if you wish (they are optional) and then finally to do the "pack". GP then goes to work and puts all of this into the assigned GRAM. It keeps track of the amount of GRAM used, so that you can do more if there is room, and in fact you are asked if you wish to do another. If you do, the whole process is repeated. Finally when you indicate you are finished, you may save the GRAM to disk (equivalent to Save Module on the GK, so you haven't lost anything if you don't do it now) and then the program quits.

If however, instead of inserting the whole program into GRAM, you only wish the loader that will bring it off your chosen device, the process is slightly different. The instruction manual was unfortunately rather murky here, and necessitated a "HAALP" call from me to Peter. There was rather a lot of explanation of how the appropriate files work, and little on how to use them. HERE is what you must do. The "filename" to load is DSK1.EA5;S;L or DSK1.EA5;S;H The first loads into low memory, the second into the top of high memory. You should choose the one that is NOT overwritten by the program file you wish loaded (see my article in the last issue of TopIcs to figure out where your program loads). Files that load at >A000 should work with either loader, although I discovered that, for reasons I still don't understand, PTERM which does load at >A000, would not work with the low memory loader). You then provide the same information for menu name, subprogram name, and DSR name, as before. However when the file is loaded there is a flag that then makes the program prompt you for "additional filename". This is where you insert the information for which program is to be loaded - you must make a permanent choice for device location, e.g. DSK1., or in my case RD., so that you might type in DSK1.UTIL1 or whatever you want. I suggest that you make a utility

disk with all the programs that you will have loaded from the menu, and keep it in the appropriate drive while you are working, except when you need another disk there. Of course, it does not have to be in drive #1. Be careful that your utility disk IS there when you choose that item from the menu - this version of GP has no error checking, and the computer will lock up if there is an error.

The method to load modules is the same except the loader program is called GK;S It will provide you with the same prompt for an "additional filename". These module and program loaders are very short, so that there is room for lots of them in one GRAM. You may therefore very well run out of room on your menu screen - see the next article for various alterations that may or must be made to your operating system to allow for more than nine items on the menu. I have one additional quibble with this FINE program. If you make a single mistake in setting up a series of loaders for the menu, the whole GP will crash, and you must start over. Similarly, if it turns out that one of your loaders doesn't work, as I found out with PTERM, you must also start over.

The GRAM PACKER is an incredible value at $19.00 (available from the Club). Final grade:

PERFORMANCE:	A
EASE OF USE:	A-
DOCUMENTATION:	B
VALUE:	A+
FINAL GRADE:	A

2. OPERATING SYSTEM MODS

Several modifications have to be made to your operating system in GRAM 0 in order to make full use of the GRAM PACKER. You will be using your Gram Kracker Editor to accomplish these (option #5 from the GK main menu). Rather than describe all the keystrokes each time, I will remind you of the general method here. First of all, when you get to the editor screen, press FCTN 1 once to get to GRAM memory. Now when you are instructred to search for a string, press FCTN 5 for search. The cursor will be on the "start" address. Accept the default of 0000 if it is there, or type it in, then press enter to get to "finish" and type 2000. Now press FCTN 9 and type in your search string, remembering FCTN = to get to hex if that is what you are searching for (in general it will be). Back up the cursor one space to get it over the last character in the search string then press enter. If the string is not found, the edit field will not change - if it is found, the address in the upper left hand corner will reflect the location of the first byte of the string found. Now press FCTN 5 again to get out of SEARCH, then FCTN 9 to edit, and type in the appropriate changes. You will need WP off in order to type in the changes - remember to turn it back on when you are finished typing.

The following set of changes need to be made only if you will wind up with more than 9 items on your main menu. There would be two problems if the changes were not made: 1) you wouldn't see any after 9 because of the double spacing! and 2) even if you could the key presses would be : ; < = > etc. some of which would actually be two keys (SHIFT and key). We will therefore enable single spacing on the main menu (thanks to Craig Miller in The Smart Programmer for this information) and change the sequence of key presses from numbers beginning with 1 to letters beginning with A.

First, to change to double spacing: Search for (hex) A3 52 00 3A. In many consoles this will be at 02E0. Change the 3A to 1A. Next comes a problem of another routine using temporary storage where we will need it (not actually involved with the double spacing, but needed if there ARE more than 9 items for the menu). Leaving the start and finish addresses the same, get back to SEARCH by pressing FCTN 5, FCTN 9 and type in 00 02 28 60 for the search string. You should find it at about 0380. FCTN 5 to get back to the memory window. The top line should read:

00 02 28 60 00 D6 28 AA 43 95 D2 29

change the 3rd, 7th, and 12th bytes:

00 02 40 60 00 D6 40 AA 43 95 D2 41

You should also insert the small capital character set into the TITLE SCREEN Characters using the NEWCHARS program on the original GK utility disk, otherwise the characters will touch each other top to bottom and be almost impossible to read. Note that you can only have 16 items on the menu if you are preserving TI Basic in GRAMS 1-2 because the start address is destroyed by the 17th item. I believe it is possible to use the 17th if you are using GRAMS 1-2 for other purposes, such as all of these programs!

Now to change the key presses to letters - this is simpler. First change your start address back to 0000, then search for BE 58 30. You should find it at about 0275. Change the 30 to 40. Next search for A6 75 31 (should be at 02FC) and change the 31 to 41. You will now see letters instead of numbers on the main menu.

I found another problem with many programs: they do no bother to change the keyboard unit to be scanned, assuming it to be 5, since that is where the E/A module is when option #5 is chosen. The problem is that the operating system is using keyboard unit 3 at the time the menu is set up (for this reason you can use lower case letters for the key press on the menu - they will be converted to upper case). Here is a simple fix: 12 bytes

past the 41 you just typed in you should see 06 03 Cx A4, where the x is probably a E. Change the first three to 05 18 00. Now FCTN 9 to get out of memory window, move the cursor to the address after the g in the upper left hand corner, and type in 1000. Now FCTN 9 again, press enter to "home" the cursor, and type in the following: 06 03 Cx BE 80 C6 02 05 03 0y where x is the same as you just found above, and y is 3 higher (in hex) than the address where you found the 06 03 (if that was 030A as it was in my console, then y would be D). This changes the keyboard unit to 5.

For those of you using SBUG6, as I do often, and who wish to use it from the main menu, you may have found that the small character set is not loaded, which is a PAIN! It's OK if you have loaded it from E/A #5. Here is a fix: it incorporates MG's GPLLNK inserted directly into memory and then a simple BLWP @GPLLNK DATA >004A and then return to the beginning of the actual program. You will need a sector editor for this. First find the FDR of the file (catalog sector). Change byte 16(>10) from 92 to EA. Second find the first actual data sector of the file. Change byte 3 from 92 to EA and bytes 24-25(>18-19) from 62 84 to 7D D2. Finally go to the LAST sector of the file (there are 30 data sectors) and starting at byte 146(>92) carefully type in the following over whatever is there:

```
7D 7E 7D 9E 7D C2 17 6C 00 50 00 00
00 00 00 00 00 00 C8 1B 83 EB C8 3E
83 EC C3 20 20 0E C8 00 20 0E 02 E0
83 E0 06 94 C9 20 7D 92 83 02 05 E0
83 73 04 60 00 60 C1 20 16 6C 06 94
02 E0 7D 7E C8 0C 20 0E 03 00 02 00
0B 00 C8 00 83 4A 04 20 7D 8C 00 4A
04 60 62 84
```

3. XBASIC PROGRAMS DIRECT FROM THE MENU

I once asked Craig whether it was possible to run XBasic programs directly off the menu, as MSAVE does with Basic programs. The answer was no, and essentially that is true, at least as far as having them run directly from GRAM is concerned, since the XML instruction needed exists only in Basic. But I kept on thinking that if XBasic can load a program called LOAD automatically from drive #1, why can't it do others as well! What follows is my method for doing this - it is rather cumbersome since it involves typing code directly into GRAM, no program like GRAM PACKER to do it for you. At least it is rather short! The method involves the following concept: when XBasic starts up, it does a certain amount of housekeeping, and then inserts the string DSK1.LOAD into the crunch buffer in VDP ram, preceded by the length byte >0B and followed by byte >00, and then "pretends" that you typed it in with RUN, and runs it. It turns out that this area is never touched by the housekeeping chores, and hence can be done right at the start. Thus my method involves inserting the program name of your choice there instead, and setting up proper code to make an additional item on the menu. If the program isn't there, you get the same result as XBasic if LOAD isn't in drive 1 - just the "ready" prompt.

First we need to do a little patching of XBasic so that DSK1.LOAD isn't pushed in over what we will put into VDP. Go to your GK memory editor and press FCTN 1 to get to GRAM, then type in 6300 for the memory address. Now FCTN 9 to get to memory window. You should see 06 64 8E. Disable the W/P switch and type in 95 over the 8E. This bypasses the move of DSK1.LOAD to VDP. (NOTE: I am using Version 110, I hope all the addresses are the same for you!) Next examine location 6006-6007. This should contain 6338 which is a pointer to the first application program (and last in the case of the pure XB module). Now move to 6338, or whatever you found and look at the first 5 bytes. They will conform to TI's standard for application programs. The first two will point to the next application program (in this case 00 00 because there aren't any, but you will eventually replace them because you are creating one) the the next two refer to the start address for this program, and should be 6372 in this case. You will be changing this. the fifth is the length byte for the following text and the text is what will go on the main menu screen. We will use the same set-up for our own application program.

You will now have to decide WHERE you are going to place the code to be described below. If you are using a pure XB module, you can use 7800 since 7800-7FFF is free. I am using Danny Michael's combination XB/EA and so this space is used. There is free space in GRAM 7, starting at F5CE however. Replace the 6372 with the start address you are using. Now go to the address you have selected and type in the following:

```
31 00 0B A8 20 63 51 05 63 72
```

If you did nothing more you would still have the same functioning XB module, because the code you just typed in performs the move of the text DSK1.LOAD and then brances back to where XB normally starts. Now the fun begins. The next address, which is F5D8 will be the location of the next application program, so go back and type it in over the 00 00 at 6338, indicating that there WILL be more. Now decide on the menu name for your program and determine the length of the name (in hex of course). I believe the maximum allowed for the actual name is 18 (>12). Now go back to F5D8 and type 00 00 00 00 xx "text" where xx is the length byte you just determined, and "text" is the actual text for the menu.

Next determine the length of the devicename.filename you wish to have loaded, e.g. DSK4.MENULOAD has a length of >D. Directly after the text for the menu you have just typed in, type this new length byte then the devicename.filename. YOU MUST FOLLOW THE DEVICENAME WITH A 00!!! Note down the address where the length byte of the filename is located AND the address just following the 00 (I will call these ADD1 and ADD2). Now type in the following code:

31 00 yy A8 20 ADD1 05 63 72

where ADD1 is the TWO bytes address just determined, and yy is the length of the filename PLUS TWO. Finally go back to the application program header and type in the two bytes of ADD2 over the 3rd and 4th 00's.

As an example, if the program to be loaded was named MENULOADER and was on DSK4 and your title for the menu was MISC. PROGRAMS, then the code beginning at F5CE should look like this:

```
>F5CE 31 00 00 A8 20 63 51 05 63 72 00 00
>F5DA F5 FC 0E 4D 49 53 43 2E 20 50 52 4F
>F5E6 47 52 41 4D 53 0F 44 53 4B 34 2E 4D
>F5F2 45 4E 55 4C 4F 41 44 45 52 00 31 00
>F5FE 11 A8 20 F5 EB 05 63 72
```

You would then follow the same general rules if you wanted to add more programs to your menu. Now at one key press the program DSK4.MENULOADER would load and run.

By the way, here is a short program allowing you to set up all your favorite programs to run without typing in the names: you merely insert them in the DATA statement, and follow the last with a "" If you save this program on your utility disk under the name you used in the above auto load then you will quickly get a menu of these programs when you press the "MISC. PROGRAMS" key and be able to pick your program with one more key press. This way you can still have the auto load of DSK1.LOAD for use with programs that need it. For this program to run properly you MUST type in line 170 first, exactly as written!

```
100 DATA RD.PRO1,RD.PRO2,""!ADD MORE IF YOU WISH
110 CALL CLEAR
120 X=X+1 :: READ A$(X):: IF A$(X)<>"" THEN 120
130 DISPLAY AT(1,1)BEEP:"PRESS FOR" :: FOR Y=1 TO
X-1 :: DISPLAY AT(2*Y+1,2):Y;"  ";A$(Y):: NEXT Y
140 CALL KEY(0,K,S):: IF S=0 THEN 140 ELSE K=K-48
150 CALL INIT :: B$=A$(K):: L=LEN(B$):: CALL LOAD(
-45,L+4):: CALL LOAD(-42,L)
160 FOR X=1 TO L :: CALL LOAD(X-42,ASC(SEG$(B$,X,1
))):: NEXT X :: CALL LOAD(X-42,0)
170 RUN "0123456789ABCDEF"
```

```
THOMAS S. FREEMAN
HOME    COMPUTER

PRESS FOR

A FOR FUNNELWRITER
B FOR MYARC DISK MANAGER
C FOR LOAD TX6
D FOR LOAD EA/XB
E FOR DISK+AID
F FOR PTERM
G FOR MASS TRANSFER
H FOR DISKASSEMBLER
I FOR RAPIDCOPY
J FOR SKIP & CTRL CODES
K FOR MENU
L FOR GK EXTENDED BASIC
M FOR EDITOR/ASSEMBLER
```

```
1 !HERE IS A SPACE FILLER.THIS PROGRAM WILL PRINT ALL DV/80 FILES ON A DISK,CATALOG FIRST,NEW PAGE TO START EACH FILE
100 OPTION BASE 1
110 DIM A1$(127),DV80(127)
120 OPEN #2:"PIO" :: PRINT #2:CHR$(27)&"N"&CHR$(2);!SKIP OVER PERF
130 CALL CLEAR
140 FOR L=1 TO 5 :: READ T$(L):: NEXT L
150 DATA DIS/FIX,DIS/VAR,INT/FIX,INT/VAR,PROGRAM
160 DISPLAY AT(12,1):"DRIVE #1" :: ACCEPT AT(12,8)SIZE(-1)VALIDATE("1234")BEEP:D$
170 OPEN #1:"DSK"&D$&".",INPUT ,RELATIVE,INTERNAL :: INPUT #1:A$,J,J,K
180 PRINT #2:"DSK";D$;" - DISKNAME= ";A$:"AVAILABLE=";K;"USED=";J-K:" FILENAME  SIZE    TYPE    P"
190 PRINT #2:"---------- --- ---------- -"
200 FOR L=1 TO 127 :: INPUT #1:A1$(L),A1,J1,K1 :: IF ABS(A1)=2 AND K1=80 THEN DV80(L)=1
210 IF LEN(A1$(L))=0 THEN 270
220 PRINT #2:A1$(L);TAB(12);J1;TAB(17);T$(ABS(A1));:: IF ABS(A1)=5 THEN 240
230 B$=" "&STR$(K1):: PRINT #2:SEG$(B$,LEN(B$)-2,3);
240 IF A1>0 THEN PRINT #2 :: GOTO 260
250 PRINT #2:TAB(28);"Y"
260 NEXT L
270 CLOSE #1 :: PRINT #2:CHR$(12);
280 FOR L=1 TO 127 :: IF A1$(L)="" THEN CLOSE #2 :: STOP ELSE IF DV80(L)=0 THEN 310
290 OPEN #1:"DSK"&D$&"."&A1$(L),INPUT :: PRINT #2:CHR$(14);A$;".";A1$(L)
300 IF EOF(1)THEN PRINT #2:CHR$(12);:: CLOSE #1 :: GOTO 310 ELSE LINPUT #1:B$ :: PRINT #2:B$ :: GOTO 300
310 NEXT L
```

MERRY CHRISTMAS FROM THE LIBRARY

FOR THE GREAT SUPPORT YOU HAVE GIVEN TO THE **LIBRARY** THIS PAST YEAR THE BOARD MEMBERS WISH TO GIVE YOU A CHRISTMAS PRESENT OF **1/2 PRICE** SALE FOR ANY PROGRAM IN OUR LIBRARY FOR THIS MONTH DECEMBER. THIS 50% DISCOUNT APPLIES TO LA99 USER GROUP MEMBER ONLY.

SPECIAL DISK OF THE MONTH

210 CALENDAR $5.00 2-SSSD DISK (448) 10 DIFFERENT WAYS TO PRINT A CALENDAR. BIG, SMALL, SMALLER, TINY, APPOINTMENT, PLANNING, BANNER, POSTER, ETC FOR ANY YEAR OR ANY MONTH.

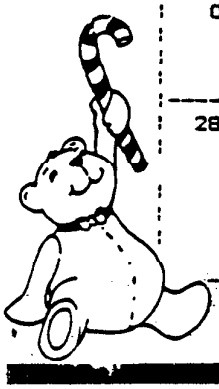
THE FOLLOWING CALENDAR IS A SAMPLE OF ONE OF THESE PROGRAMS. MAKE A SPECIAL CALENDAR FOR YOUR FAMILY

LIBRARIAN FRED MOORE 7730 EMERSON AVE. L.A. CA 90045 213-670-4293

LA 99 GROUP
1984

DECEMBER

SUN	MON	TUE	WED	THU	FRI	SAT
1/2 PRICE MONTH TO ALL MEMBERS	1 nm moon	2 BOARD MEETING 7:00PM	3	4	5	6
7	8 fq moon	9	10 HAPPY BIRTHDAY GEORGE STEFFEN	11	12	13
14 ACP COMPUTER SWAPMEET AMAHEIM	15 HAPPY BIRTHDAY STEVE CHALCRAFT	16 full moon	17 HAPPY BIRTHDAY FLETCHER WICKER	18 REGULAR MEETING 7:PM PARTY BRING GIFT	19 HAPPY BIRTHDAY MARGARET HUTTON	20
21 FIRST DAY OF WINTER	22	23	24 CHRISTMAS EVE no meeting REMEMBER SANTA lq moon	25 MERRY CHRISTMAS FROM THE LIBRARY	26	27 CHANUKAH
28	29	30	31 NEW YEAR EVE DONT DRINK & DRIVE nw moon	HAPPY BIRTHDAY FRED MOORE NEW YEAR DAY	1/2 PRICE MONTH TO ALL MEMBERS	



CLUB MEETINGS

Los Angeles 99er Computer Group: Fourth Wednesday of each month, 7:15 PM, at Torrance Library, 3031 Torrance Blvd., Torrance.

Pomona Valley 99ers Computer Group: Second Tuesday of each month, 7:00 PM, at Cable Airport Cafe, 13th & Benson, Upland. Call Joy Warner, 982-9971, nights.

San Fernando Valley 99er Computer Group: Second ~~Tuesday~~ WEDNESDAY of each month, 7:30 PM, Doctors' Conference Room, Sherman Oaks Community Hospital, 4929 Van Nuys Blvd., Sherman Oaks.

San Gabriel Valley 99/4 Users Group: First Wednesday of each month at West Covina Public Library, 1601 W. Covina Parkway, West Covina.

Users Group of Orange County: Third Thursday of each month, 7:30 PM at Westchester Community Service Center (1 block east of Beach Blvd.), Jackson and Westminster, Westminster, CA.

Executive Board meeting of the Los Angeles 99ers is held at 7:15 PM, the first Tuesday of each month at Merit Savings Bank, 18501 S. Western Ave., Torrance. ** ALL MEMBERS ARE INVITED **

The first line on the address label shows the last issue you will receive for members or the last issue received by us for exchanges.

Board Meetings for the LA 99ers is held on the 1st Tuesday of each month at MERIT SAVINGS at 18501 Western Ave. Gardena
The meeting starts at 7:15. *** EVERYONE IS INVITED ***

XXXXXXXXXXXXXXXXXXXXXXXXXX
FIRST CLASS
XXXXXXXXXXXXXXXXXXXXXXXXXX

Exchng * Oct 86
Miami County Area 99/4A UG
P. O. Box 1194
Peru, IN 46970

RENEW YOUR MEMBERSHIP ... SEND IN YOUR CHECK TODAY!

LA 99ers Computer Group
P.O. Box 3547
Gardena, CA 90247-7247

99'er West-Fest '87